

IMECE2011-62489

MASSIVELY PARALLEL COMPUTATIONAL FLUID DYNAMICS WITH LARGE EDDY SIMULATION IN COMPLEX GEOMETRIES

Andrew Duggleby, Joshua L. Camp

Mechanical Engineering, Texas A&M University
and Exosent, LLC
College Station, TX

Yuval Doron

Exosent, LLC
College Station, TX

Paul F. Fischer

Mathematics and Computer Science
Argonne National Laboratory
Argonne, IL

ABSTRACT

To perform complex geometry large eddy simulations in an industrially relevant timeframe, one must reduce the total time to half a day (overnight simulation). Total time includes the time of developing the mesh from the computer-aided design (CAD) model and simulation time. For reducing CAD-to-mesh time, automatic meshing algorithms can generate valid but often non-efficient meshes with often up to an order of magnitude more grid points than a custom-based mesh. These algorithms are acceptable only if paired with high-performance computing (HPC) platforms comprising thousands to millions of cores to significantly reduce computational time. Efficient use of these tools calls for codes that can scale to high processor counts and that can efficiently transport resolved scales over the long distances and times made feasible by HPC. The rapid convergence of high-order discretizations makes them particularly attractive in this context. In this paper we test the combination of automatic hexahedral meshing with a spectral element code for incompressible and low-Mach-number flows, called Nek5000, that has scaled to $P > 262,000$ cores and sustains $> 70\%$ parallel efficiency with only ≈ 7000 points/core. For our tests, a simple pipe geometry is used as a basis for comparing with previous fully resolved direct numerical simulations.

INTRODUCTION

In an engineering design cycle, design analysis must be quick in order to evaluate and modify a design to the desired performance. In some cases, reduced analyses or empirical correlations provide a good enough initial evaluation to proceed to

prototype; otherwise, one must perform computational analyses to test design validity. The field of computer-aided design (CAD) and computational analysis has had significant growth over the last decade. SolidWorks, a leading CAD software product, reports 1.5 million users globally [1]. Stress analyses, vibration analyses, and dynamics analyses are all possible with most commercially available CAD software. Fluid mechanics analyses (or computational fluid dynamics, CFD), however, has only recently begun to be incorporated into CAD software. This situation is due mainly to the numerical difficulties and stability concerns that have kept CFD from automatic, or “pushbutton,” solutions.

For most other stress-based solid mechanics analyses, the finite-element numerical representation of conservation of mass, momentum, and energy results in an unconstrained minimization problem (also known as the Rayleigh-Ritz criterion) [2, 3]. In fluid mechanics, because of pressure that couples the conservation of momentum with conservation of mass, the numerical solution is a constrained minimization problem. For example, in incompressible fluid flow, only velocity solutions that are divergence free are valid. These stability difficulties also persist in low-Mach (slightly compressible) flows, where thermal pressure is linked to a state equation (ideal gas) [4], and in transonic and supersonic flows, where only nonoscillatory solutions near shocks are allowed [5, 6]. Solving the time-dependent fluid equations as opposed to the time-averaged equations (as done in most solvers) is significantly more stable because the solution is integrated forward in time from a valid solution. However, this approach is often prohibitive and requires typically an order of magnitude or more increase in computing, as well as robust models for any unresolved scales (subgrid stress models or turbulence

models) [7–9]. Therefore, the first challenge in reducing the time for a CFD solution is reducing the simulation time [10].

The second challenge in shortening the total time of CFD is meshing. In all CFD, mesh quality plays a significant role in improving both solution convergence as well as accuracy [11]. Furthermore, having resolution only in regions where resolution is needed, either through adaptive meshing or custom hand-built meshing, also provides a significant reduction in simulation time. However, meshing often takes the longest time, often an order of magnitude longer to perform than the actual simulation [12]. Thus, an automated meshing tool is needed that generates a high-quality mesh rapidly.

Automatic Hexahedral Meshing

Because of the significant role that the mesh of a given domain plays in a simulation's quality, the creation and improvement of meshing algorithms continue to be a major focus of research. Three common criteria used to compare algorithms are the time needed to create the mesh, the mesh quality on the interior of the domain, and the mesh quality near the boundaries [12]. For fluid simulations, the mesh quality near the boundary can play a significant role, particularly for simulations with high turbulence. Fully hexahedral meshes exhibit many desirable traits over meshes with mixed elements or fully tetrahedral elements, such as improved efficiency and a better representation of the boundaries [13, 14]. However, there exist no universal, robust meshing algorithms for fully hexahedral meshes, although a few methods have shown promise.

One such method is the grid-based algorithm [15]. This method begins with a structured grid (quads for a 2D domain, hexes for a 3D domain) inside the domain to be meshed. New elements are created to fill any gaps between the original mesh and the boundaries. This algorithm produces elements with excellent quality for the domain interior, but the mesh quality suffers near the boundary. Other methods attempt to automate the process one would go through when creating a mesh manually; the domain is decomposed into pieces that can be easily meshed [16–19]. Still, other methods attempt to create a mesh beginning at the boundary and moving inward, such as plastering [20] and whisker weaving [21]. These methods produce meshes with good quality at the boundaries, but the mesh quality can suffer at the interior (if the mesh can be created at all).

Massively Parallel Spectral Element Algorithms

We have recently undertaken a series of spectral-element based simulations in complex domains for turbomachinery applications, including a high-pressure turbine (HPT) heat transfer calculation and a low-pressure turbine (LPT) aerodynamic calculation. Despite decades of advancement, these complex flows remain not fully understood because of their complex turbulence dynamics and thus nonoptimal designs. To change this situa-

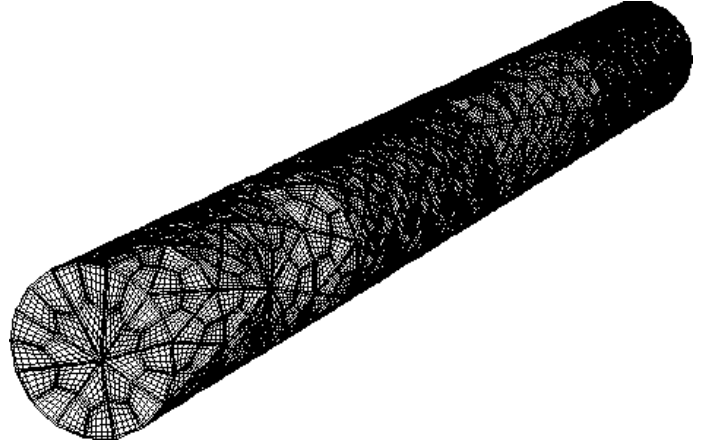


Figure 1: Automatically generated hexahedral mesh from a pipe CAD geometry, resulting in $E=4556$ elements. With polynomial order $N=9$, total grid points is 3.3 million.

tion requires the use of the inherently more accurate large eddy simulation (LES) in the design cycle. This is a challenging task because HPT and LPT cases involve complex geometries and high element count to resolve the flow features and, including a complex turbulence generation section, curved blade geometries, boundary layers, and in the HPT case cooling holes. We hypothesize that using a massively parallel LES algorithm (greater than 1,000 CPUs) with an automatically generated mesh has the potential to perform these simulations in an industrially relevant timeframe.

In this paper we test this hypothesis in a simple yet well-studied turbulent pipe flow geometry. An automatic hexahedral meshing routine algorithm from Exosent, LLC, is used to mesh a simple pipe CAD geometry, and Nek5000, a massive parallelization of a spectral-element algorithm, is used for solving the time-dependent (and thus more stable) fluid equations. A large eddy simulation using the automatic mesh shown in Fig. 1 is run and the results are compared with those of a direct numerical simulation (DNS) on a structured, hand-built mesh shown in Fig. 2 from the same solver.

NUMERICAL SETUP

The flow simulations are based on Nek5000, a spectral-element solver developed over the past 20 years for turbulence research [22–29]. Nek5000 solves the Navier-Stokes equations, which represent conservation of mass, momentum, and energy

$$\frac{\partial U_j}{\partial x_j} = 0, \quad (1)$$

$$\left(\frac{\partial}{\partial t} + U_j \frac{\partial}{\partial x_j} \right) U_i = -\frac{1}{\rho} \frac{\partial}{\partial x_i} P + \nu \frac{\partial^2}{\partial x_j \partial x_j} U_i + \frac{\partial \tau_{ij}^{SGS}}{\partial x_j}, \quad (2)$$

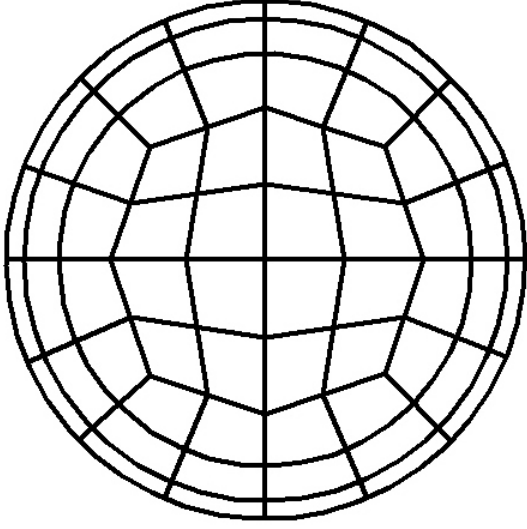


Figure 2: Cross section of mesh for direct numerical simulation turbulent pipe from Duggleby et al. [29] with $E = 2560$ elements (64 in cross section), and 4.4 million grid points using polynomial order $N=12$.

with velocity U_i , pressure P , kinematic viscosity ν , and for LES the subgrid stress tensor τ_{ij}^{SGS} . Nek5000 employs a geometrically flexible yet exponentially convergent spectral-element discretization in space, dividing the computation into E elements over which the solution is represented as a high-order (usually $N=7-15$) Lagrangian interpolant, for a total of $n \approx EN^3$ grid-points. Time discretization is based on high-order operator splitting methods that yield independent substeps for advection, dissipation, and incompressibility. Advection is treated explicitly in time while the viscous and pressure substeps are solved implicitly with, respectively, Jacobi-preconditioned conjugate gradients and multigrid-preconditioned GMRES [22, 24].

The data localization of the spectral-element method allows for minimal communication between elements, resulting in efficient parallelization. Moreover, the use of tensor-product-based operator evaluation results in memory demands that are equivalent to 7-point finite difference stencils. The computational complexity scales as $O(nN)$ but the leading order work term is cast in the form of highly efficient matrix-matrix products that place minimal demand on memory bandwidth [30]. Currently the solver has shown sustained performance of 19% of peak on 260,000 processors on Julich BG/P supercomputer with over 70% parallel efficiency [31].

Direct Numerical Simulation with Custom Mesh

Using DNS to solve the conservation of mass and momentum equations for all possible temporal and spatial scales comes at a cost. The computational operations scale as Re_τ^4 , and storage

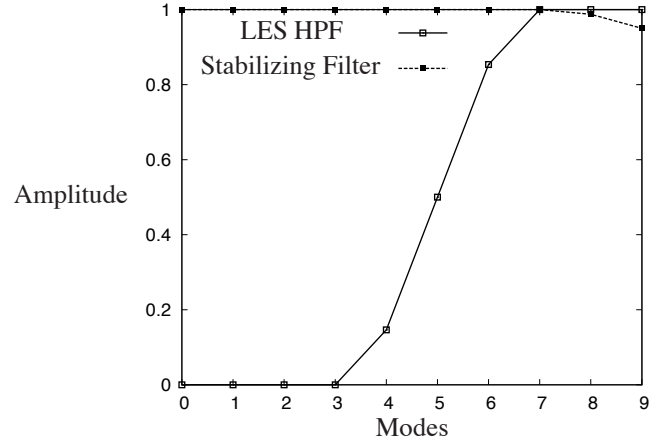


Figure 3: Transfer function for the high-pass filter LES model are set similar to a cosine function that filters the large scales and leaves the small scales unfiltered. A stabilizing filter is used for all simulations (DNS and LES) where the last 2 modes are filtered.

requirements scale as $\text{Re}_\tau^{9/4}$ [32].

The DNS turbulent pipe flow data are generated by using Nek5000 and a custom built hexahedral mesh, shown in Fig. 2. The mesh has $E = 2560$ elements, and the data were generated with polynomial order $N = 12$ for a total of 4.4 million grid points. The flow is driven by a mean streamwise pressure gradient to obtain a Reynolds number of $\text{Re}_\tau = 150$. When nondimensionalized with the mean velocity U_m , the Reynolds number is $\text{Re}_m = U_m D / \nu \approx 4300$. The domain length is $L = 20R = 10D$ (diameters) or $z^+ = 3000$ in wall units (distance nondimensionalized by the length ν/U_τ), which is long enough to use periodic boundary conditions to achieve realistic turbulent inlet and outlet conditions. The mesh is structured, shown in Fig. 2, using a gridding preprocessor associated with Nek5000. Data are acquired for 2,100 samples every $t^+ = 8$. This corresponds to a total simulation time of $t^+ = U_\tau^2 t / \nu = 16800$ viscous time units, which is roughly $tU_m/D \approx 800$. Further details can be found in Duggleby et al. [29].

Large Eddy Simulation with Automatic Mesh

When DNS is too computationally expensive or would take too long to reach a solution, large eddy simulation is an option. For LES, not all spatial scales are resolved, and the effect of the unresolved scales are modeled through a sub-grid stress term τ_{ij}^{SGS} in Eq. 2. The model used here is a high-pass-filtered (HPF) Smagorinsky model from Stolz et al. [33]. In this model, the SGS stress term is based off a HPF velocity-strain rate rather than the

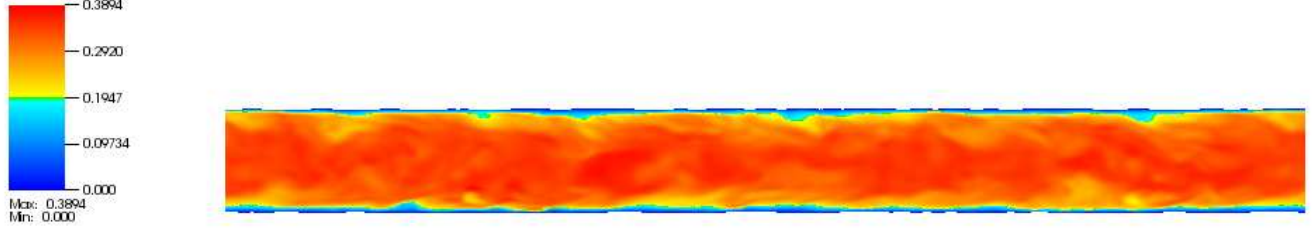


Figure 4: Contour plot of velocity magnitude in a 2D slice for the automatic mesh pipe simulation with the subgrid stress (SGS) model.

full velocity-strain rate:

$$\tau_{ij}^{SGS} - \frac{\delta_{ij}}{3} \tau_{kk}^{SGS} = -2\nu_t^{HPF} S_{ij}, \quad (3)$$

$$S_{ij} = \frac{1}{2} (\partial U_i x_j + \partial U_j x_i), \quad (4)$$

$$\nu_t^{HPF} = (C_s^{HPF} \Delta)^2 |S_{ij}(\tilde{U})|, \quad (5)$$

where Δ is the filter width taken as the largest distance between the Gauss-Lobatto-Legendre (GLL) points in physical space, C_s^{HPF} is a constant with a value 0.1, and \tilde{u} is the high pass filtered velocity field. A dynamic version of the model with varying C_s^{HPF} is possible; however, the fixed coefficient has shown satisfactory results [33, 34] and is thus chosen for the present investigation. This model performs well in transitional flows as well as nonisotropic flows [34, 35]. The model is a good candidate for a generalized LES model that requires no parameter tweaking or adjustment for any given problem. The filter is shown in Fig. 3. Nek5000 also has a stabilizing filter developed by Fischer et al. [36] for the spectral element basis function that filters out the high-frequency components, shown in Fig. 3. The filter is applied after each time step and preserves interelement continuity [37]. This avoids the nonphysical spectral build up in the energy spectra that is usually found in underresolved DNS or LES. The HPF Smagorinsky model is applied after each time step and acts upon the velocity components filtered by the stabilizing filter.

For the LES, the mesh is generated with a CAD hexahedral meshing algorithm from Exosent, LLC, shown in Fig. 1, with $E = 4556$ elements. Adding a boundary layer resolution to an automatically generated mesh is a planned extension of this work. The mesh is imported into Nek5000 and run at the same Reynolds number $Re_\tau = 150$. Polynomial order $N=9$ was used, totaling 3.3 million grid points. The domain is $21R$ long, one diameter longer than the DNS. Periodic boundary conditions at $z = 0$ are enforced via a recycling plane at $z = 20R$ downstream from the inlet, with an outlet (stress-free) boundary condition is prescribed at $z = 21R$. Initial conditions are the same as the DNS flow. The velocity field at the inlet is scaled such that the flow

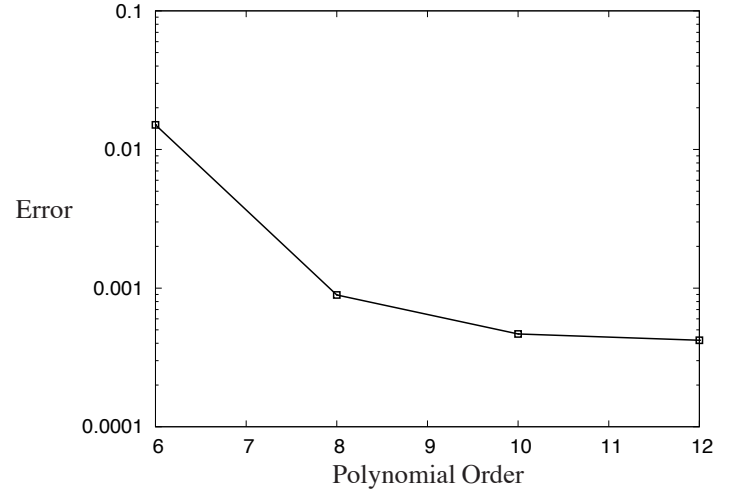


Figure 5: Convergence plot without SGS model shows the beginning of convergence. With the custom mesh from Duggeby et al. [29], the solution was already converging exponentially at 10^{-3} error with the same amount of grid points.

rate is constant, set to match the same value as the DNS, using a proportional controller.

RESULTS

An instantaneous slice of the automated mesh flow field is shown in Fig. 4. Five domain flow throughs were simulated from the initial conditions (including transition). Statistics were averaged over approximately one flow-through. A grid convergence plot shown in Fig. 5 for the simulation without the LES SGS model reveals convergence although high error due to the coarseness of the mesh as polynomial order is increased. With the custom mesh where resolution is added where most needed near the wall, the error is already to 10^{-3} [29]. With the LES SGS model, increasing the polynomial order changes the filter width because it is coupled to the grid spacing when the number of filtered modes is kept constant. This is similar to eddy dif-

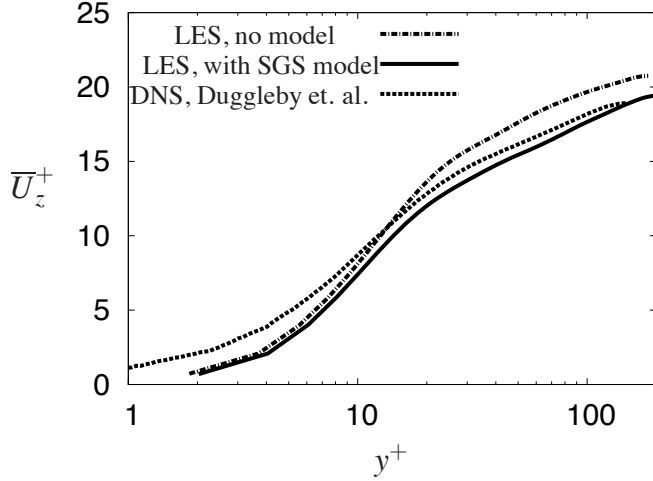


Figure 6: Mean axial velocity \overline{U}_z^+ versus y^+ for the LES with SGS model (solid), without SGS model (dot-dashed) and DNS from Dugleby et al. [29] (dashed). Because the automated mesh is only piecewise smooth on the wall, the mean is not fully defined to $r = R$. With LES model the error is approximately 5%.

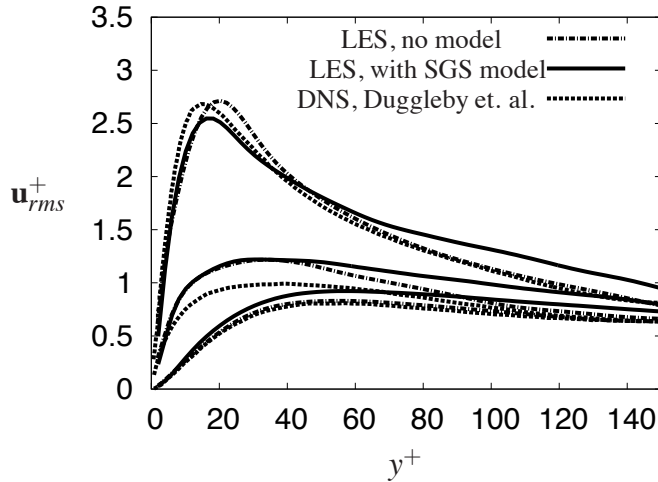


Figure 7: Velocity fluctuation rms versus y^+ for the LES with SGS model (solid), without SGS model (dot-dashed) and DNS from Dugleby et al. [29] (dashed). The u_{rms}^+ peak location for the LES with SGS model is in close agreement with the DNS peak. Both LES overshoot azimuthal and streamwise rms due to the roughened wall but are within 15%.

fusivity LES on finite-volume meshes. Estimating convergence via grid independent solution is thus prohibited, as pointed out by Celik et al. (2009) [38]. Since no convergence study can be presented, validation against experimental values becomes even more important to ensure simulation confidence.

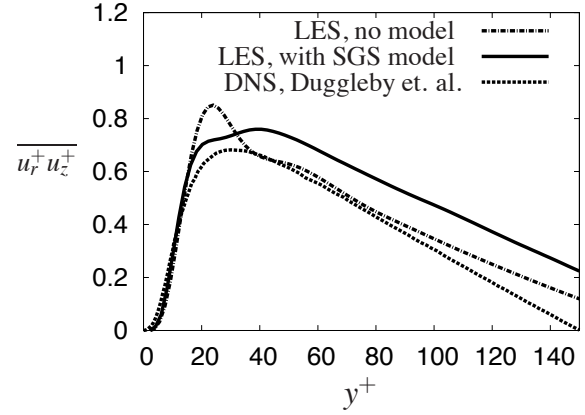


Figure 8: Reynolds shear stress $\overline{u_r^+ u_z^+}$ versus y^+ for the LES with SGS model (solid), without SGS model (dot-dashed) and DNS from Dugleby et al. [29] (dashed). The LES with SGS model is closer to the DNS, but both overshoot due to the roughened wall. Away from the wall ($y^+ > 50$) the slope of the LES with SGS model matches the DNS.

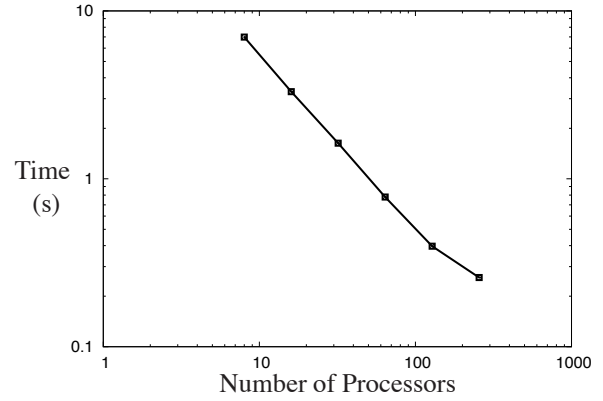


Figure 9: Automatic mesh simulation showing strong scaling up to 256 processors, resulting in a simulation time of 12.5 hours.

The automatic computational mesh is only piecewise cylindrical, and so the walls are not smooth. This extra roughness creates higher wall shear stress, resulting in an Re_τ of approximately 200 and 185 for the LES with and without the SGS model. Smoothing the walls is a planned extension of this work. The mean axial velocity is shown in Fig. 6. Note that because of the piecewise continuous approximation, data is not fully defined all the way to $y^+ = (R - r)u_\tau/\nu = 0$ as R is constant as defined from the CAD geometry. Similarly, the standard turbulence near-wall result of $\overline{U}_z^+ \approx y^+$ also does not hold. With the SGS model, the mean velocity is within 5% error.

Plots of root-mean-square (rms) of fluctuating velocity components $u_{r,rms}^+, u_{\theta,rms}^+, u_{z,rms}^+$ are shown in Fig. 7, where u_i is the

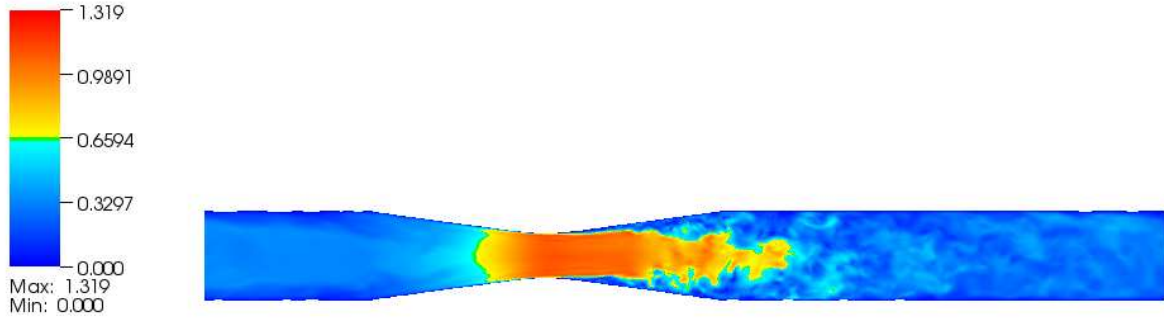


Figure 10: Extending to more complicated geometries, a Venturi converging-diverging nozzle is added to the end of the pipe geometry at $z = 21R$. The geometry is sketched in CAD, mesh generated, and LES results completed overnight. Shown are instantaneous contours of velocity magnitude in a 2D cross-section.

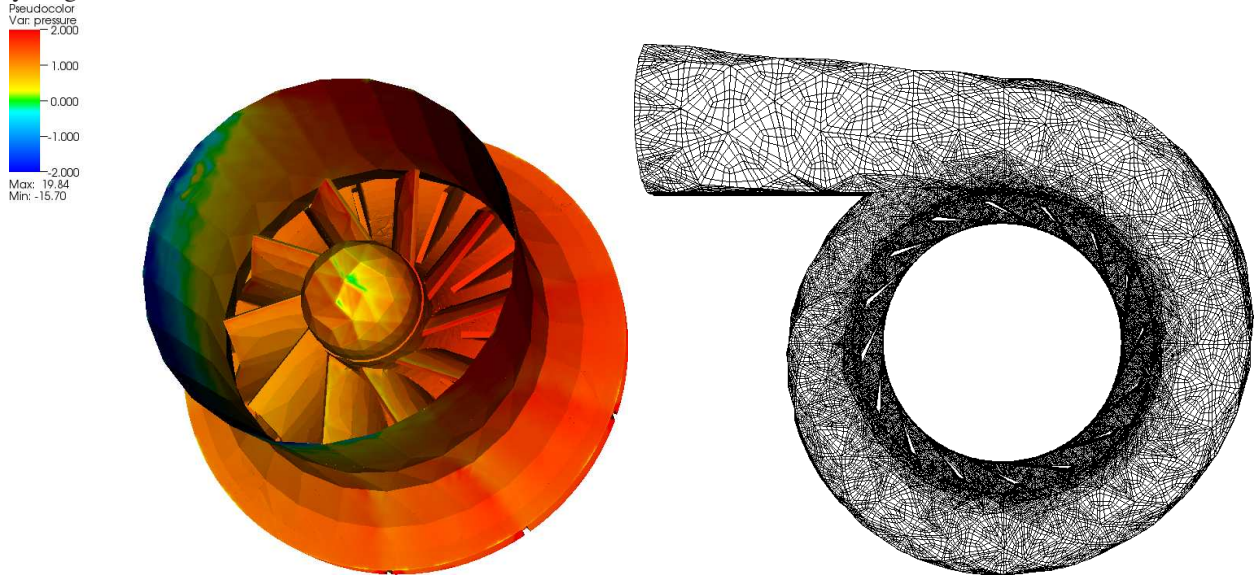


Figure 11: Extending to even more complicated geometries, a centrifugal compressor (left) with initial pressure isosurfaces on the walls is shown, and the mesh (showing GLL points) of the corresponding valute (right).

fluctuating component of the velocity field $U_i = \overline{U}_i + u_i$ and \overline{U}_i denotes an average over time as well as azimuthal and stream-wise planes. The $u_{r,rms}^+$ peak with the LES model is closer to the DNS peak. LES both with and without the SGS model overshoots the DNS $u_{\theta,rms}^+, u_{z,rms}^+$ components by 15%, most likely because of the roughened wall. For the same reason both also overshoot the Reynolds stress $\overline{u_r^+ u_z^+}$ shown in Fig. 8, but the LES model with SGS is closer to the DNS model.

CPU scaling for this study is shown in Fig. 9. A couple of flow-throughs take only 3 hours and so fit within the criteria of an overnight simulation. Along with a mesh generation that took under five minutes, this is a valid example.

Extension to generalized geometries

As an extension to more complicated geometries, a Venturi converging-diverging nozzle was added to the CAD geometry after $z = 21R$. The exact curvature was “hand drawn” in CAD and not an empirical formula. The new mesh again was generated in under 5 minutes, and the simulation ran overnight. A flow visualization 2D cross-section of axial velocity is shown in Fig. 10. An even more complex extension to a centrifugal compressor is shown in Fig. 11 with initial pressure isosurface results. Again, the time from CAD to mesh was under 5 minutes.

CONCLUDING REMARKS

An automatic CAD to mesh algorithm paired with a massively parallel Navier-Stokes algorithm is tested for industrial relevance. This relevance is measured by a 12-hour maximum time requirement from CAD to numerical solution, representing an overnight simulation. Although not restricted to simple geometries, this approach is tested here in a simple pipe geometry with high-quality DNS data generated from the same algorithm to compare against. The LES model used in the study is based on a high-pass-filtered strain rate that has shown good performance in complex geometries. The LES results match qualitatively very well with previous DNS results, with only small errors in turbulent peak locations and 5-15% error in turbulent statistics. Total simulation time was ~ 12.5 hours from CAD to LES results.

ACKNOWLEDGEMENTS

This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Dept. of Energy, under Contract DE-AC02-06CH11357.

NOMENCLATURE

Uppercase

C_s	Smagorinsky constant
D	Pipe diameter
E	Elements
N	Polynomial order
P	Pressure
R	Pipe radius
S_{ij}	Strain rate tensor
U_m	Mean streamwise velocity
Re	Reynolds number

Lowercase

\tilde{u}	HPF velocity field
r	Radial coordinate
u_τ	Shear velocity, $\sqrt{\tau_w/\rho}$
u_i	Velocity vector
x_i	Position vector
z	Streamwise (axial) coordinate

Greek

Δ	LES filter width
δ	Kronecker delta
θ	Azimuthal coordinate
ν	Kinematic viscosity
ρ	Density
τ_{ij}	Stress tensor
τ_w	Wall shear stress

Superscripts

+	Wall unit normalization
---	-------------------------

Acronyms

CAD	Computer aided design
-----	-----------------------

CFD	Computational fluid dynamics
DNS	Direct numerical simulation
GLL	Gauss-Lobatto-Legendre
HPF	High-pass filtered
LES	Large eddy simulation
RMS	Root-mean-square
SGS	Subgrid stress

REFERENCES

- [1] Dassault Systèmes SolidWorks Corp., 2011. Solidworks fact sheet, 2011 Q1. <http://www.solidworks.com>.
- [2] Doering, C. R., 2009. "The 3d Navier-Stokes problem". *Annual Review of Fluid Mechanics*, **41**(1), pp. 109–128.
- [3] Bochev, P. B., and Gunzburger, M. D., 2009. "Least-squares finite element methods". In *Applied Mathematical Sciences*, S. S. Antman, J. E. Marsden, and L. Sirovich, eds., Vol. 166. Springer.
- [4] Tomboulides, A. G., Lee, J. C. Y., and Orszag, S. A., 1997. "Numerical simulation of low mach number reactive flows". *Journal of Scientific Computing*, **12**(2), pp. 139–167.
- [5] Pirozzoli, S., 2011. "Numerical methods for high-speed flows". *Annual Review of Fluid Mechanics*, **43**(1), pp. 163–194.
- [6] Leveque, R. J., 1992. "Numerical methods for conservation laws". In *Lectures in Mathematics, ETH Zürich*. Birkhäuser Verlag.
- [7] Menzies, K., 2009. "Large eddy simulation applications in gas turbines". *Phil. Trans. Roy. Soc. A*, **367**(1899), pp. 2827–2838.
- [8] Sagaut, P., and Deck, S., 2009. "Large eddy simulation for aerodynamics: status and perspectives". *Phil. Trans. Roy. Soc. A*, **367**(1899), pp. 2849–2860.
- [9] Hutton, A., 2009. "The emerging role of large eddy simulation in industrial practice: challenges and opportunities". *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **367**(1899), pp. 2819–2826.
- [10] Cant, S., 2002. "High-performance computing in computational fluid dynamics: progress and challenges". *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, **360**(1795), pp. 1211–1225.
- [11] Löhner, R., 2007. *Applied Computational Fluid Dynamics Techniques: an introduction based on finite element methods*, 2nd ed. Wiley & Sons.
- [12] Tautges, T. J., 2001. "The generation of hexahedral meshes for assembly geometry: survey and progress". *International Journal for Numerical Methods in Engineering*, **50**, pp. 2617–2642.
- [13] Cifuentes, A., and Kalbag, A., 1992. "A performance

- study of tetrahedral and hexahedral elements in 3d element structural analysis". *Finite Elements in Analysis and Design*, **12**, pp. 313–318.
- [14] Pakal, D., Seshadri, M., Canann, S., and Saigal, S., 1998. A comparative study of hexahedral and tetrahedral elements. Report, Sandia National Laboratories.
- [15] Schneiders, R., 1996. "A grid-based algorithm for the generation of hexahedral element meshes". *Engineering with Computers*, **12**, pp. 168–177.
- [16] Price, M., and Armstrong, C., 1995. "Hexahedral mesh generation by medial surface subdivision: Part I: Solids with convex edges". *International Journal for Numerical Methods in Engineering*, **38**, pp. 3335–3359.
- [17] Price, M., and Armstrong, C., 1997. "Hexahedral mesh generation by medial surface subdivision: Part II: Solids with flat and concave edges". *International Journal for Numerical Methods in Engineering*, **40**, pp. 111–136.
- [18] Li, T., McKeag, R., and Armstrong, C., 1995. "Hexahedral meshing using midpoint subdivision and integer programming". *Computer Methods in Applied Mechanics and Engineering*, **124**, pp. 171–193.
- [19] Holmes, D., 1995. "Generalized method of decomposing solid geometry in hexahedron finite elements". In Proc. of the 4th International Meshing Roundtable, SAND95-2130.
- [20] Stephenson, M., Canann, S., and Blacker, T., 1992. Plastering: a new approach to automated, 3d hexahedral mesh generation. Report, Sandia National Laboratories, SAND89-2192.
- [21] Tautges, T. J., 1996. "The whisker weaving algorithm: a connectivity-based method for constructing all-hexahedral finite element meshes". *International Journal for Numerical Methods in Engineering*, **39**, pp. 3327–3349.
- [22] Fischer, P. F., Lottes, J. W., Pointer, D., and Siegel, A., 2008. "Petascale algorithms for reactor hydrodynamics". In J. Phys. Conf. Ser. 125012076.
- [23] Fischer, P. F., Ho, L. W., Karniadakis, G. E., Ronquist, E. M., and Patera, A. T., 1988. "Recent advances in parallel spectral element simulation of unsteady incompressible flows". *Comput. & Struct.*, **30**, pp. 217–231.
- [24] Lottes, J. W., and Fischer, P. F., 2004. "Hybrid multi-grid/Schwarz algorithms for the spectral element method". *J. Sci. Comp.*, **24**(1), pp. 45–78.
- [25] Tufo, H. M., and Fischer, P. F., 1999. "Terascale spectral element algorithms and implementations". In Proc. of the ACM/IEEE SC99 Conf. on High Performance Networking and Computing, IEEE Computer Soc. Gordon Bell Prize paper.
- [26] Fischer, P. F., and Lottes, J. W., 2003. "Hybrid Schwarz-multigrid methods for the spectral element method: Extensions to Navier-Stokes". In Proc. of the 15th Int. Conf. on Domain Decomposition Methods.
- [27] Duggeby, A., and Paul, M., 2010. "Computing the Karhunen-Loève dimension of an extensively chaotic flow field given a finite amount of data". *Computers & Fluids*, **39**, pp. 1704–1710.
- [28] Duggeby, A., Ball, K. S., and Schwaenen, M., 2009. "Structure and dynamics of low reynolds number turbulent pipe flow". *Phil. Trans. Roy. Soc. A*, **347**, pp. 473–488.
- [29] Duggeby, A., Ball, K. S., Paul, M. R., and Fischer, P. F., 2007. "Dynamical eigenfunction decomposition of turbulent pipe flow". *J. of Turbulence*, **8**(43), pp. 1–24.
- [30] Deville, M. O., Fischer, P. F., and Mund, E. H., 2002. "High-order methods for incompressible fluid flow". In *Cambridge Monographs on Applied and Computational Mathematics*, P. G. Ciarlet, A. Iserles, R. V. Kohn, and M. H. Wright, eds., Vol. 9. Cambridge University Press.
- [31] S. Kerkemeier, S. Parker, P. F. F., 2010. Scalability of the NEK5000 spectral element code. Jülich Blue Gene/P Extreme Scaling Workshop 2010, technical report.
- [32] Pope, S. B., 2000. *Turbulent Flows*. Cambridge University Press.
- [33] Stolz, S., 2005. "High-pass filtered eddy-viscosity models for large-eddy simulations of compressible wall-bounded flows". *Journal of Fluids Engineering*, **127**(4), pp. 666–673.
- [34] Schwaenen, M., Meador, C., Camp, J., Jagannathan, S., and Duggeby, A., 2011. "Massively-parallel direct numerical simulation of turbine vane endwall horseshoe vortex dynamics and heat transfer". *ASME Paper GT2011-45915*.
- [35] Jagannathan, S., Schwänen, M., and Duggeby, A., 2011. "Low pressure turbine relaminarization bubble characterization using massively-parallel large eddy simulations". *J. Fluids Engineering*. in review.
- [36] Paul Fischer, J. M., 2001. "Filter-based stabilization of spectral element methods". *C.R.Acad. Sci. Ser. I-Anal Numer.*, **163**, pp. 193–204.
- [37] Fischer, P. F., Kruse, G. W., and Loth, F., 2002. "Spectral element methods for transitional flows in complex geometries". *J. Sci. Comput.*, **17**(1-4), pp. 81–98.
- [38] Celik, I., Klein, M., and Janicka, J., 2009. "Assessment measures for engineering LES applications". *J. Fluids Eng.*, **131**(031102).

The submitted manuscript has been created [in part] by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.